In [3]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [4]:
```python
# Read the CSV file
df = pd.read_csv("C:\\Users\\srini\\Desktop\\DataSet\\2020.csv")
```

In [5]:
```python
# Set display precision
pd.set_option("display.precision", 2)
```

In [6]:
```python
# Display the first 3 rows
print(df.head(3))
```

```
  Country name Regional indicator  Ladder score  \
0      Finland     Western Europe          7.81
1      Denmark     Western Europe          7.65
2  Switzerland     Western Europe          7.56

   Standard error of ladder score  upperwhisker  lowerwhisker  \
0                            0.03          7.87          7.75
1                            0.03          7.71          7.58
2                            0.04          7.63          7.49

   Logged GDP per capita  Social support  Healthy life expectancy  \
0                  10.64            0.95                     71.9
1                  10.77            0.96                     72.4
2                  10.98            0.94                     74.1

   Freedom to make life choices  Generosity  Perceptions of corruption  \
0                          0.95       -0.06                       0.20
1                          0.95        0.07                       0.17
2                          0.92        0.11                       0.30

   Ladder score in Dystopia  Explained by: Log GDP per capita  \
0                      1.97                              1.29
1                      1.97                              1.33
2                      1.97                              1.39

   Explained by: Social support  Explained by: Healthy life expectancy  \
0                          1.50                                   0.96
1                          1.50                                   0.98
2                          1.47                                   1.04

   Explained by: Freedom to make life choices  Explained by: Generosity  \
0                                        0.66                      0.16
1                                        0.67                      0.24
2                                        0.63                      0.27

   Explained by: Perceptions of corruption  Dystopia + residual
0                                     0.48                 2.76
1                                     0.50                 2.43
2                                     0.41                 2.35
```

In [7]:
```python
# Find the maximum value of 'Ladder score' and corresponding entry
max_ladder_score = np.max(df["Ladder score"])
max_ladder_score_entry = df.iloc[df['Ladder score'].argmax()]
print(f"Maximum Ladder score: {max_ladder_score}")
print("Corresponding entry:")
print(max_ladder_score_entry)
```

```
Maximum Ladder score: 7.808700085
Corresponding entry:
Country name                                         Finland
Regional indicator                            Western Europe
Ladder score                                            7.81
Standard error of ladder score                          0.03
upperwhisker                                            7.87
lowerwhisker                                            7.75
Logged GDP per capita                                  10.64
Social support                                          0.95
Healthy life expectancy                                 71.9
Freedom to make life choices                            0.95
Generosity                                             -0.06
Perceptions of corruption                                0.2
Ladder score in Dystopia                                1.97
Explained by: Log GDP per capita                        1.29
Explained by: Social support                             1.5
Explained by: Healthy life expectancy                   0.96
Explained by: Freedom to make life choices              0.66
Explained by: Generosity                                0.16
Explained by: Perceptions of corruption                 0.48
Dystopia + residual                                     2.76
Name: 0, dtype: object
```

In [8]:
```python
# Display DataFrame info
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 153 entries, 0 to 152
Data columns (total 20 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Country name                                153 non-null    object
 1   Regional indicator                          153 non-null    object
 2   Ladder score                                153 non-null    float64
 3   Standard error of ladder score              153 non-null    float64
 4   upperwhisker                                153 non-null    float64
 5   lowerwhisker                                153 non-null    float64
 6   Logged GDP per capita                       153 non-null    float64
 7   Social support                              153 non-null    float64
 8   Healthy life expectancy                     153 non-null    float64
 9   Freedom to make life choices                153 non-null    float64
 10  Generosity                                  153 non-null    float64
 11  Perceptions of corruption                   153 non-null    float64
 12  Ladder score in Dystopia                    153 non-null    float64
 13  Explained by: Log GDP per capita            153 non-null    float64
 14  Explained by: Social support                153 non-null    float64
 15  Explained by: Healthy life expectancy       153 non-null    float64
 16  Explained by: Freedom to make life choices  153 non-null    float64
 17  Explained by: Generosity                    153 non-null    float64
 18  Explained by: Perceptions of corruption     153 non-null    float64
 19  Dystopia + residual                         153 non-null    float64
dtypes: float64(18), object(2)
memory usage: 24.0+ KB
None
```

In [9]:
```python
# Print the first country name and its regional indicator
print(df['Country name'][0])
print(df['Regional indicator'][0])
```

```
Finland
Western Europe
```

In [10]:
```python
# Display frequency of entries for each country
print(df["Country name"].value_counts().sort_values(ascending=False))
```

```
Country name
Finland                     1
Austria                     1
United States               1
Switzerland                 1
Iceland                     1
                           ..
Central African Republic    1
Rwanda                      1
Zimbabwe                    1
South Sudan                 1
Afghanistan                 1
Name: count, Length: 153, dtype: int64
```

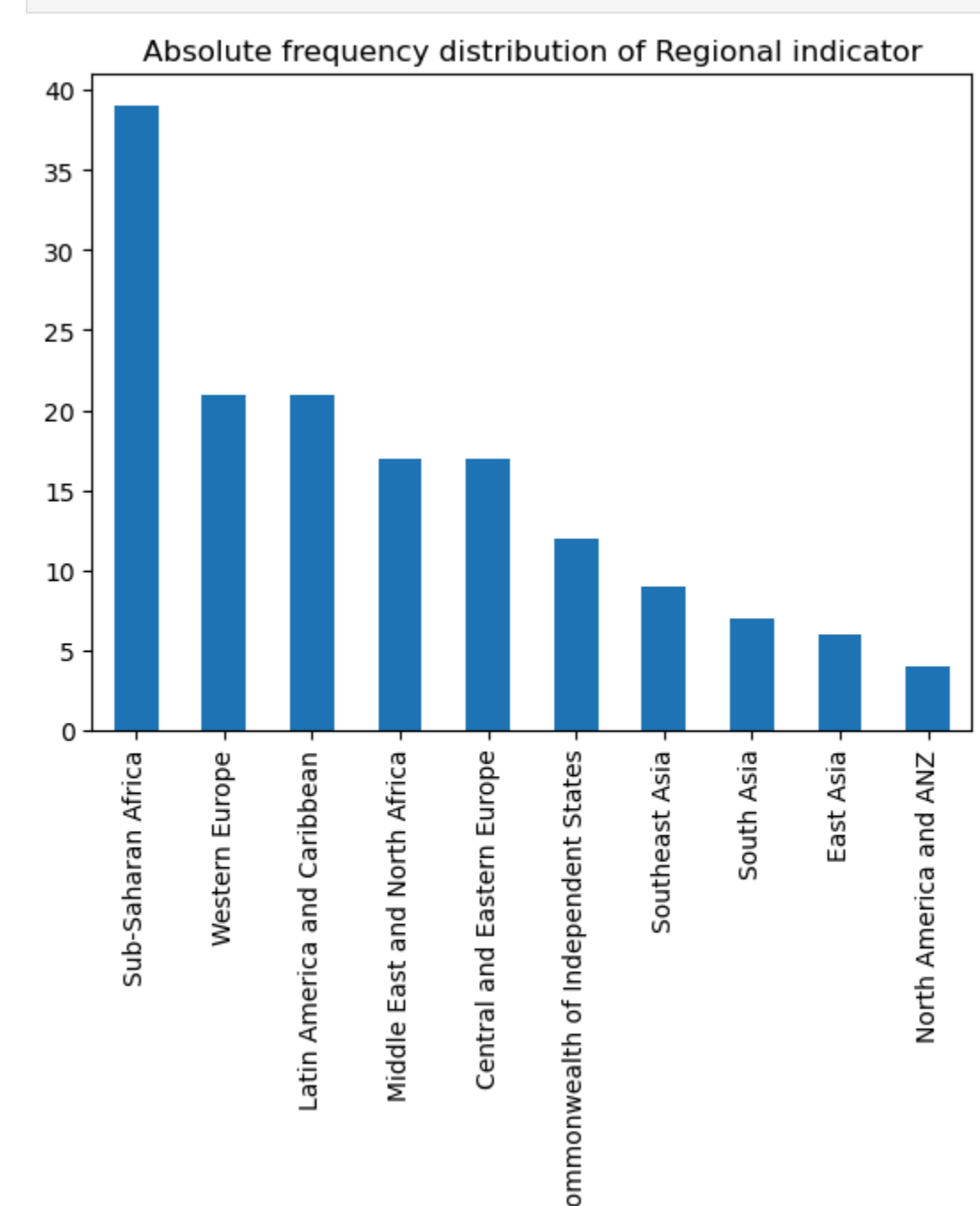
In [11]:
```python
# Display frequency of each regional indicator
print(df['Regional indicator'].value_counts())
```

```
Regional indicator
Sub-Saharan Africa                    39
Western Europe                        21
Latin America and Caribbean           21
Middle East and North Africa          17
Central and Eastern Europe            17
Commonwealth of Independent States    12
Southeast Asia                         9
South Asia                             7
East Asia                              6
North America and ANZ                  4
Name: count, dtype: int64
```

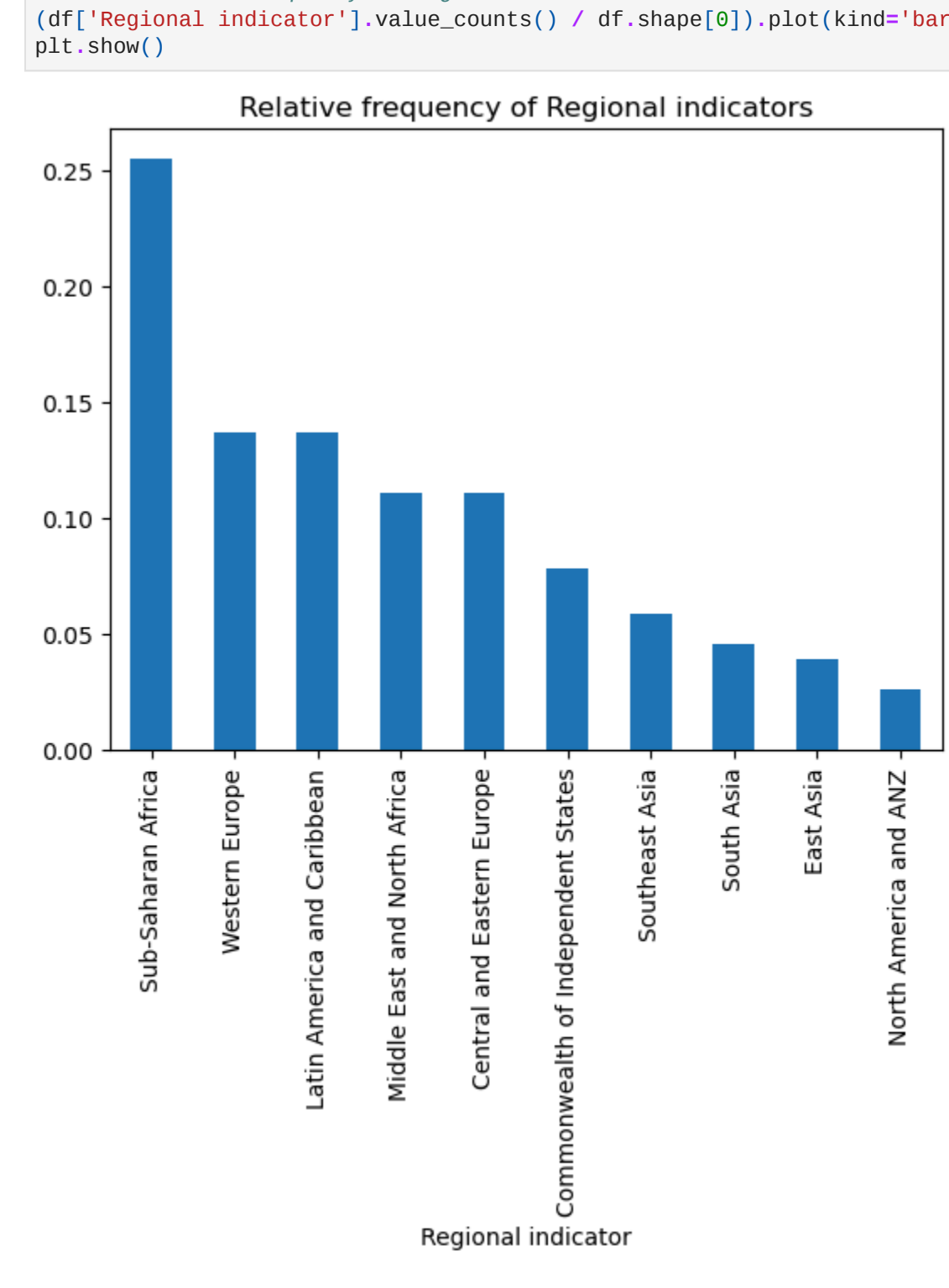
In [12]:
```python
# Number of regions in the dataset
print(f"The number of regions in our dataset is: {df['Regional indicator'].nunique()}")
```

```
The number of regions in our dataset is: 10
```

In [13]:
```python
# Plot absolute frequency distribution of regional indicators
df['Regional indicator'].value_counts().plot(kind='bar', title='Absolute frequency distribution of Regional indicator')
plt.show()
```



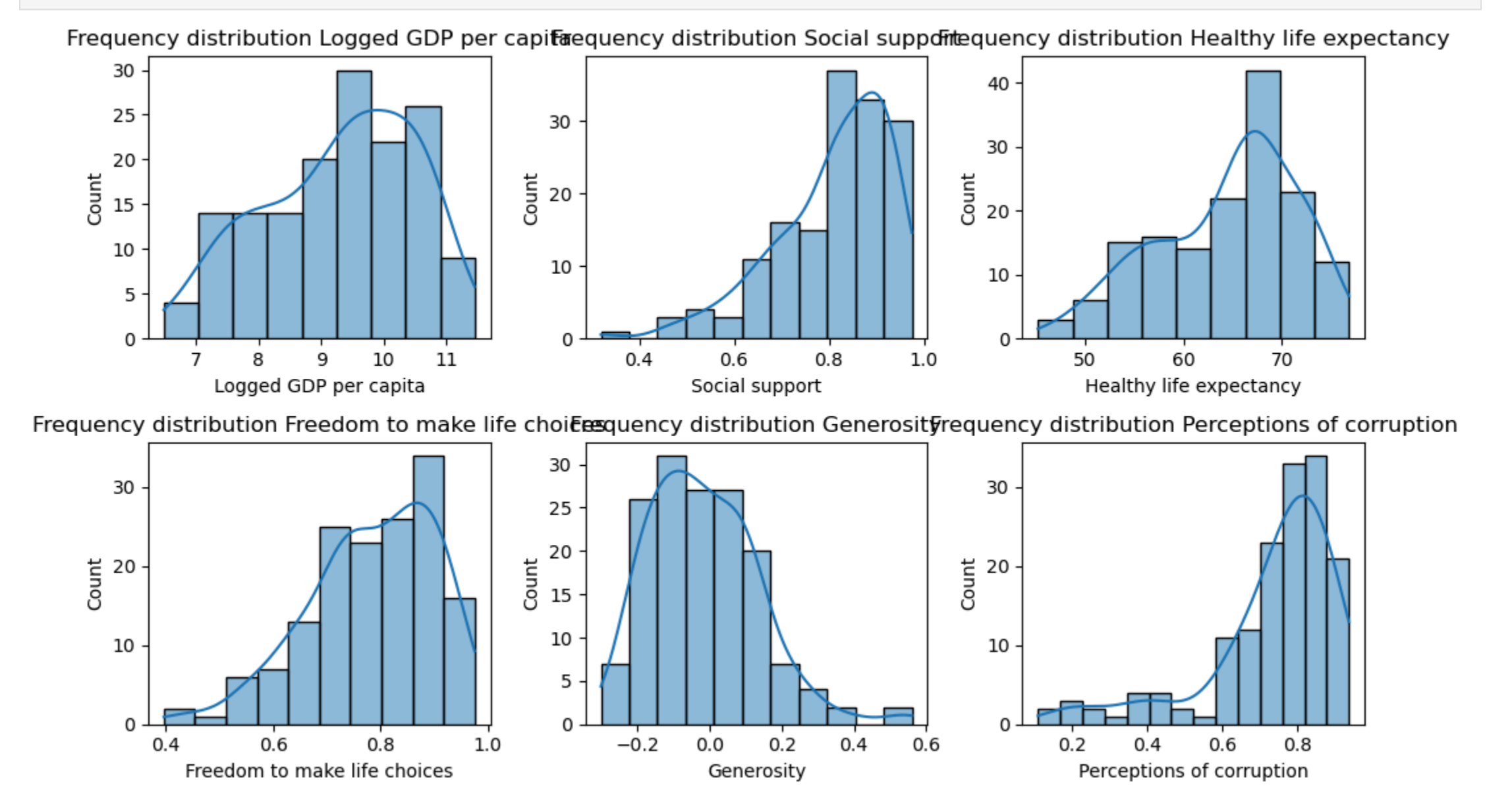
In [14]:
```python
# Plot relative frequency of regional indicators
(df['Regional indicator'].value_counts() / df.shape[0]).plot(kind='bar', title='Relative frequency of Regional indicators')
plt.show()
```



In [15]:
```python
# Descriptive statistics
print(df.describe())
```

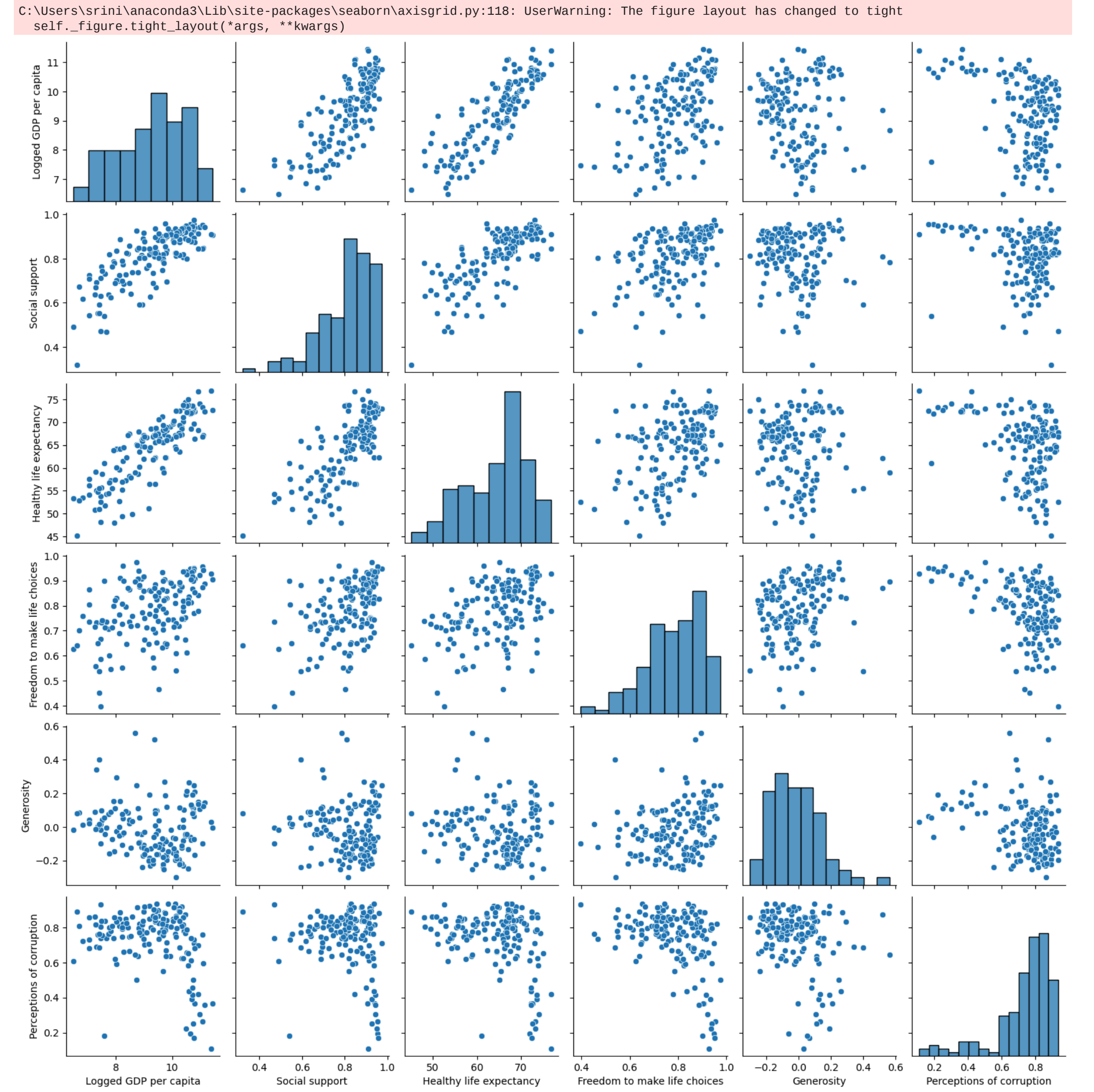
```
       Ladder score  Standard error of ladder score  upperwhisker  \
count        153.00                          153.00        153.00
mean           5.47                            0.05          5.58
std            1.11                            0.02          1.10
min            2.57                            0.03          2.63
25%            4.72                            0.04          4.83
50%            5.51                            0.05          5.61
75%            6.23                            0.06          6.36
max            7.81                            0.12          7.87

       lowerwhisker  Logged GDP per capita  Social support  \
count        153.00                 153.00          153.00
mean           5.37                   9.30            0.81
std            1.13                   1.20            0.12
min            2.51                   6.49            0.32
25%            4.60                   8.35            0.74
50%            5.43                   9.46            0.83
75%            6.14                  10.27            0.91
max            7.75                  11.45            0.97

       Healthy life expectancy  Freedom to make life choices  Generosity  \
count                   153.00                        153.00      153.00
mean                     64.45                          0.78       -0.01
std                       7.06                          0.12        0.15
min                      45.20                          0.40       -0.30
25%                      58.96                          0.71       -0.13
50%                      66.31                          0.80       -0.03
75%                      69.29                          0.88        0.09
max                      76.80                          0.97        0.56

       Perceptions of corruption  Ladder score in Dystopia  \
count                     153.00                  1.53e+02
mean                        0.73                  1.97e+00
std                         0.18                  1.56e-15
min                         0.11                  1.97e+00
25%                         0.68                  1.97e+00
50%                         0.78                  1.97e+00
75%                         0.85                  1.97e+00
max                         0.94                  1.97e+00

       Explained by: Log GDP per capita  Explained by: Social support  \
count                            153.00                        153.00
mean                               0.87                          1.16
std                                0.37                          0.29
min                                0.00                          0.00
25%                                0.58                          0.99
50%                                0.92                          1.20
75%                                1.17                          1.39
max                                1.54                          1.55

       Explained by: Healthy life expectancy  \
count                                 153.00
mean                                    0.69
std                                     0.25
min                                     0.00
25%                                     0.50
50%                                     0.76
75%                                     0.87
max                                     1.14

       Explained by: Freedom to make life choices  Explained by: Generosity  \
count                                      153.00                    153.00
mean                                         0.46                      0.19
std                                          0.14                      0.10
min                                          0.00                      0.00
25%                                          0.38                      0.12
50%                                          0.48                      0.18
75%                                          0.58                      0.26
max                                          0.69                      0.57

       Explained by: Perceptions of corruption  Dystopia + residual
count                                   153.00               153.00
mean                                      0.13                 1.97
std                                       0.11                 0.56
min                                       0.00                 0.26
25%                                       0.06                 1.63
50%                                       0.10                 2.05
75%                                       0.16                 2.35
max                                       0.53                 3.44
```

In [16]:
```python
# Plot frequency distribution of selected features
columns = ['Logged GDP per capita', 'Social support', 'Healthy life expectancy', 'Freedom to make life choices', 'Generosity', 'Perceptions o
scols = int(len(columns) / 2)
srows = 2
fig, axes = plt.subplots(srows, scols, figsize=(10, 6))

for i, col in enumerate(columns):
    ax_col = i % scols
    ax_row = i // scols
    sns.histplot(df[col], kde=True, ax=axes[ax_row, ax_col])
    axes[ax_row, ax_col].set_title('Frequency distribution ' + col)

plt.tight_layout()
plt.show()
```



In [17]:
```python
# Pairplot of selected features
sns.pairplot(df[columns])
plt.show()
```

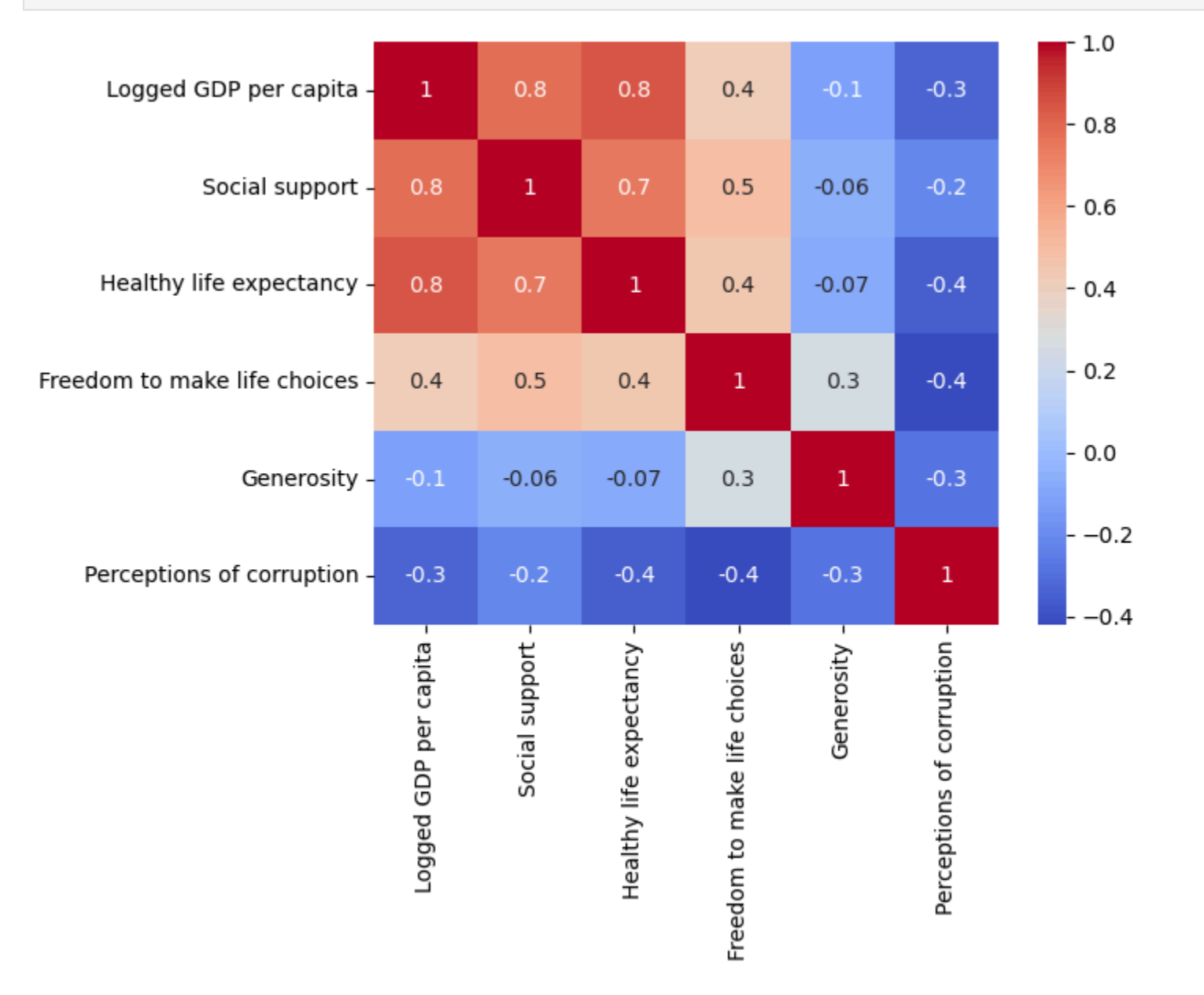
```
C:\Users\srini\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



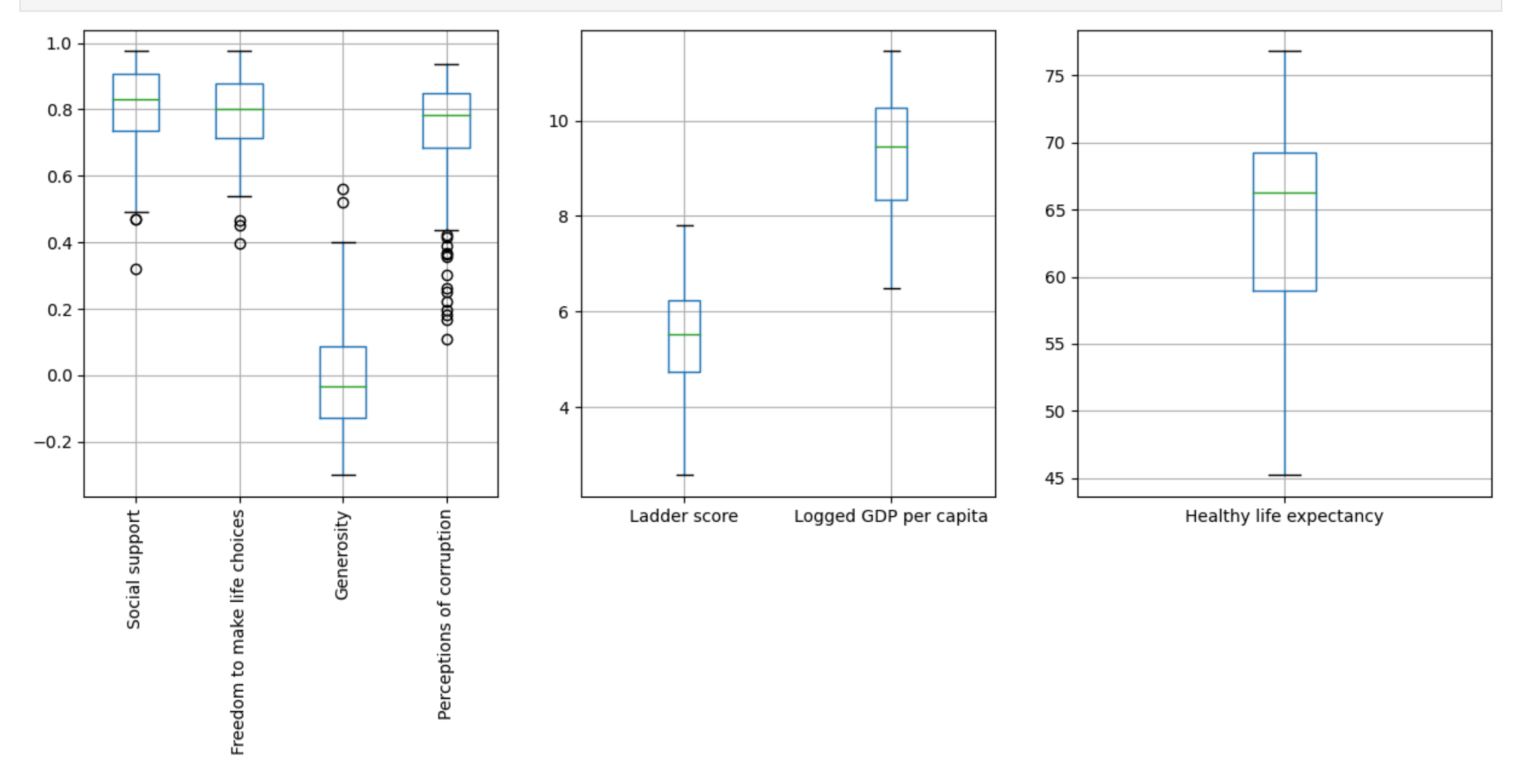
In [18]:
```python
# Pairplot of selected features colored by regional indicator
sns.pairplot(df[columns + ['Regional indicator']], hue="Regional indicator", palette="Paired")
plt.show()
```

```
C:\Users\srini\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

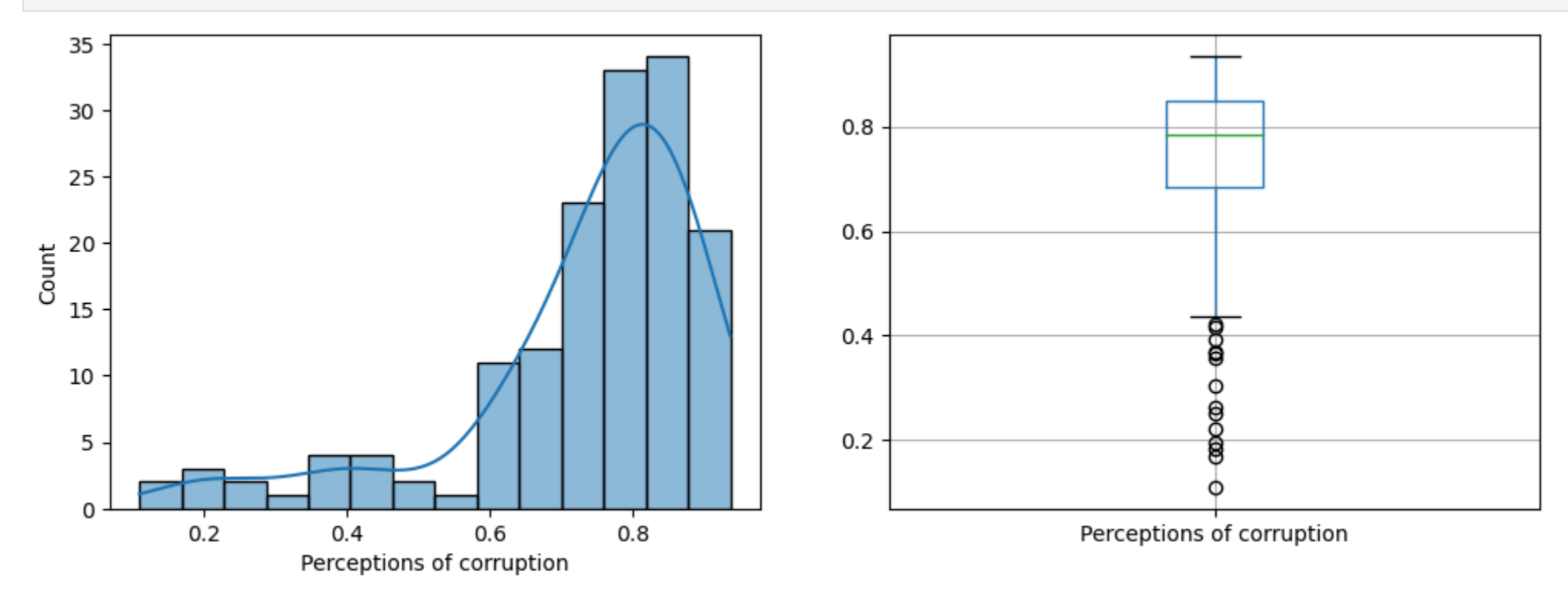
In [19]:
```python
# Heatmap of correlation between features
sns.heatmap(df[columns].corr(), annot=True, fmt='.1g', cmap='coolwarm')
plt.show()
```



In [20]:
```python
# Boxplots for different feature groups
small = ['Social support', 'Freedom to make life choices', 'Generosity', 'Perceptions of corruption']
medium = ['Ladder score', 'Logged GDP per capita']
large = ['Healthy life expectancy']

f, axs = plt.subplots(1, 3, figsize=(15, 5))
df.boxplot(column=small, ax=axs[0])
axs[0].set_xticklabels(small, rotation=90)
df.boxplot(column=medium, ax=axs[1])
df.boxplot(column=large, ax=axs[2])
plt.show()
```



In [21]:
```python
# Distribution plot and boxplot for 'Perceptions of corruption'
f, axs = plt.subplots(1, 2, figsize=(12, 4))
sns.histplot(df['Perceptions of corruption'], kde=True, ax=axs[0])
df.boxplot(column=['Perceptions of corruption'], ax=axs[1])
plt.show()
```



In [22]:
```python
# Countries with low perceptions of corruption
low_corruption_countries = df[df['Perceptions of corruption'] < 0.4][['Country name', 'Perceptions of corruption']].sort_values(by='Perceptio
print(low_corruption_countries)
```

```
    Country name  Perceptions of corruption
30     Singapore                       0.11
1        Denmark                       0.17
149       Rwanda                       0.18
0        Finland                       0.20
7    New Zealand                       0.22
6         Sweden                       0.25
4         Norway                       0.26
2    Switzerland                       0.30
15       Ireland                       0.36
5    Netherlands                       0.36
9     Luxembourg                       0.37
10        Canada                       0.39
```